ISSN: 2355-9365

Implementasi dan AnalisisAlgoritma Clustering Expectation—maximization (EM) Pada Data Tugas Akhir Universitas Telkom

Ridoy ED Sirait¹, Eko Darwianto ST., MT.², Dawam Dwi Jatmiko Suwawi, ST., MT.³

1,2,3 Fakultas Informatika Universitas Telkom, Bandung

elpewe@gmail.com

Abstrak

Kebanyakan mesin pencari yang ada saat ini pada umumnya menampilkan dokumen hasil pencarian sesuai dengan urutan dokumen (*document ranking*) tanpa mengelompokkan atau mengkategorikan dokumen sesuai dengan kemiripan dokumen. Dengan jumlah dokumen yang cukup besar akan memberikan dampak negatif bagi pengguna, yaitu dibutuhkan waktu yang relatif lama untuk memilah-milah dokumen yang sesuai dengan kebutuhan pengguna. Untuk mempermudah pengguna dalam mencari informasi pada kumpulan dokumen yang cukup besar, salah satu solusinya adalah dengan cara mengelompokkan dokumen hasil pencarian sesuai dengan *keyword* yang diinputkan oleh pengguna. Dengan adanya pengelompokan dokumen hasil pencarian ini, maka pengguna tidak perlu membuka halaman terlalu banyak karena dokumen hasil pencarian telah dikelompokkan berdasarkan kemiripan dokumen-dokumen tersebut.

Salah satu algoritma *partitional* yang dapat mengelompokkan dokumen yang belum berlabel adalah *Expectation-Maximization*, yaitu algoritma yang berfungsi untuk menemukan nilai estimasi *Maximum Likelihood* dari parameter dalam sebuah model probabilistik [2]. Ciri-ciri dari algoritma ini adalah dapat mengelompokkan dokumen yang belum berlabel atau *unlabeled data* dan juga hasil pengelompokannya akan selalu *convergence*. Dari hasil percobaan didapatkan kesimpulan bahwa algoritma EM dapat mengelompokkan dokumen hasil pencarian, hal ini dapat membantu pengguna untuk mencari dokumen yang diharapkan. Akurasi tertinggi mencapai 70% dan terendah 32.58%. Penambahan algoritma *stemming* Arifin Setiono mampu meningkatkan performansi algoritma EM hingga 10%.

Kata Kunci: Clustering, Expectation-Maximization, Unsupervised, Stemming.

Abstract

Most search engines this days generally displaying document of search results based on document order (document ranking) without grouping or categorize documents based on similarity. With a large number of documents, this will make a negative impact for users, it takes a relatively long time to sort out the documents that user needs. To facilitate the user in finding information on a large number of documents, one of the solutions is to classify document of search results according to the keywords entered by the user. With the document search results are grouped, the user does not need to open up too much pages because the document of search results have been grouped based on documents similarity.

One partitional algorithm that able to classify unlabeled documents is Expectation-Maximization, this algorithm used to find the value of Maximum Likelihood estimation of parameters in a probabilistic model [2]. The characteristics of this algorithm is able to classify documents that have not been labeled or unlabeled documents and also the results of the classification will always convergence. From the experimental results it was concluded that the EM algorithm can classify documents of search results, it can help users to search for documents they expected. The highest accuracy reaches 70% and the lowest 32.58%. The addition of stemming algorithms Arifin Setiono EM algorithm can improve performance up to 10%.

Keywords: Clustering, Expectation-Maximization, Unsupervised, Stemming

1 Pendahuluan

Kebanyakan mesin pencari yang ada saat ini pada umumnya menampilkan dokumen hasil pencarian sesuai dengan urutan dokumen (document ranking) tanpa mengelompokkan atau mengkategorikan dokumen kemiripan dokumen. Salah satu fasilitas pencarian dokumen yang masih menampilkan dokumen hasil pencarian sesuai urutan dokumen adalah repository Telkom. Dimana repository berfungsi sebagai mesin pencari dokumen tugas

akhir, jurnal, *thesis*, karya ilmiah dan dokumendokumen penelitian lainnya. Pencarian dokumen pada *repository* ini biasanya menampilkan dokumen-dokumen hasil pencarian dalam jumlah yang cukup besar. Dengan jumlah dokumen yang cukup besar akan memberikan dampak negatif bagi pengguna, yaitu dibutuhkan waktu yang relatif lama untuk memilah-milah dokumen yang sesuai dengan kebutuhan pengguna.

Untuk mempermudah pengguna dalam mencari informasi pada kumpulan dokumen yang

cukup besar, salah satu solusinya adalah dengan cara mengelompokkan dokumen hasil pencarian sesuai dengan *keyword* yang diinputkan oleh pengguna. Dengan adanya pengelompokan dokumen hasil pencarian ini, maka pengguna tidak perlu membuka halaman terlalu banyak karena dokumen hasil pencarian telah dikelompokkan berdasarkan kemiripan dokumen-dokumen tersebut.

Salah satu cara untuk mengelompokkan dokumen adalah dengan metode *Clustering*. Pengelompokan dokumen atau *Document*

Clustering adalah suatu metode yang digunakan untuk mengelompokkan dokumen-dokumen ke kelompok-kelompok dalam atau Clusters berdasarkan kemiripan dokumen. sehingga dokumen yang saling berhubungan ditempatkan pada Cluster yang sama. Ada beberapa algoritma Clustering yang dikenal yaitu partitional (Expectation-maximization, K-Means) dan hierarchical (Centroid Linkage, Single Linkage), overlapping (Fuzzy C-Means) dan hybrid [1].

Hasil pencarian dokumen adalah dokumen-dokumen yang belum berlabel dan berubah-ubah sesuai dengan keyword yang diinputkan oleh pengguna, sehingga hasil pengelompokan dokumen-dokumen tersebut juga akan berubah sesuai dengan kemiripan dokumen. Algoritma yang bisa mengatasi pengelompokan yang berubah-ubah ini adalah algoritma partitional. Dimana pada algoritma partitional, dokumen bisa merupakan anggota dari suatu kelompok atau Cluster pada suatu proses namun pada proses berikutnya dokumen tersebut bisa berpindah ke Cluster lain.

Salah satu algoritma partitional yang dapat mengelompokkan dokumen yang belum berlabel adalah Expectation-Maximization, yaitu algoritma yang berfungsi untuk menemukan nilai estimasi Maximum Likelihood dari parameter dalam sebuah model probabilistik [2]. Ciri-ciri dari adalah dapat mengelompokkan algoritma ini dokumen yang belum berlabel atau unlabeled data dan juga hasil pengelompokannya akan selalu convergence. Algoritma ini memiliki dua tahap, yaitu tahap Expectation dan tahap Maximization. Pada tahap Expectation (E-step) digunakan algoritma Naïve Bayes untuk mengelompokkan data berdasarkan model parameter. Sedangkan pada tahap Maximization (M-step) akan dilakukan peng-

update-an model parameter. Tahap *E-step* dan *M-step* terus dilakukan sampai probabilitas setiap *Cluster* mencapai *convergence* [3].

Sebelum melakukan pengelompokan dokumen diperlukan proses pre-processing, yaitu parsing. cleansing. tokenizing, stopword elimination, dan stemming. Proses ini diperlukan untuk mengurangi jumlah kata yang diproses pada saat Clustering. Pelabelan dari suatu Cluster dilakukan dengan mencari label aktual yang paling banyak muncul pada suatu Cluster, kemudian mengadopsi label tersebut sebagai label Cluster. Dimana label ini merupakan label yang sudah tersedia pada data Tugas Akhir Universitas Telkom.

2 Landasan Teori

2.1 Algoritma Expectation-maximization

Expectation-Maximation (EM) termasuk algoritma *partitional* yang berbasiskan model yang menggunakan perhitungan probabilitas, bukan jarak seperti umumnya algoritma *clustering* yang lain. Jika pada algoritma K-means, parameter utamanya adalah *centroid*, maka untuk EM parameter utamanya adalah q_{mk} dan α_k untuk mendapatkan nilai r_{nk} yaitu probabilitas dokumen n masuk ke klaster k atau probabilitas klaster k beranggotakan dokumen n [3].

Langkah-langkah algoritma EM adalah sebagai berikut:

a. Guess Model Parameter

Proses ini adalah melakukan penebakan nilai probabilitas data terhadap sebuah klaster. Langkah *guess* pertama adalah *Guess probability* data klaster sebagai *model paramet*er. Inisialisasi nilai probabilitas pada data kata dilakukan secara random/ acak [11]. Untuk probabilitas klaster, totalnya hasus selalu bernilai 1.

Table 2-1 Tabel Guess Model Parameter

Y	X1				
(Klaster)		X2	X3	X4	P(Y)
0	0,1	0,3	0,8	0,8	0,7
1	0,2	0,3	0,1	0,1	0,2
2	0,7	0,4	0,1	0,1	0,1

Dimana pada tahap ini akan ditebak nilai parameter q_{mk} dan α_k

 $r_{nk} = \frac{\alpha_k (\prod_{t_m \in d_n} q_{mk}) (\prod_{t_m \notin d_n} (1 - q_{mk}))}{\sum_{k=1}^K \alpha_k (\prod_{t_m \in d_n} q_{mk}) (\prod_{t_m \notin d_n} (1 - q_{mk}))}$ (2.1)

Dimana:

 r_{nk}adalah nilai probabilitas setiap dokumen nterhadap masing-masing *cluster* atau nilai probabilitas *cluster* k terhadap sebuah dokumen.

$$\alpha_k (\prod_{t_m \in d_n} q_{mk}) (\prod_{t_m \neq d_n} (1 - q_{mk}))$$

adalah probabilitas total term terhadap sebuah klaster

 $\sum_{k=1}^{K} \alpha_k (\prod_{t_m \in d_n} q_{mk}) (\prod_{t_m \notin d_n} (1 - q_{mk}))$

adalahnilai total probabilitas semua termterhadap semua klaster.

Setelah r_{nk}didapat, maka akan dihitung Frequency Counts

$$\sum_{n=1}^{N} r_{nk} I(t_m \in d_n)$$
Maximization Step

$$q_{mk} = \frac{\sum_{n=1}^{N} r_{nk} I(t_m \in d_n)}{\sum_{n=1}^{N} r_{nk}}$$
(2.2)

Dimana:

- q_{mk}adalah nilai probabilitas terhadap sebuah klaster dimana term m tersebut merupakan anggota dari suatu dokumen n.
- $\sum_{n=1}^{N} r_{nk} I(t_m \in d_n)$ adalah frequency Counts, probabilitas klaster k terhadap semua dokumen yang mempunyai term m sebagai anggotanya (nilai term m = 1).
- $\sum_{n=1}^{N} r_{nk}$ adalah Probabilitas sebuah cluster k terhadap semua dokumen.

Kemudian dihitung probabilitas sebuah klaster

$$\alpha_k = \frac{\sum_{n=1}^{N} r_{nk}}{N} \tag{2.3}$$

Dimana N adalah probabilitas total klaster Secara singkat, langkah-langkah pengelompokan dokumen dengan algoritma Unsupervised EM adalah:

- Guess Initial Model Parameter 1.
- Compute Expected Frequency Given
 - Probabilitas total term terhadap sebuah

$$\alpha_k (\prod_{t_m \in d_n} q_{mk}) (\prod_{t_m \notin d_n} (1 - q_{mk}))$$

Frequency Counts

$$\sum_{n=1}^{N} r_{nk} I(t_m \in d_n) \tag{2.5}$$

- 3. Find MLE given Expexted Frequency
 - Nilai probabilitas term m terhadap sebuah klaster dimana term m tersebut merupakan anggota dari suatu dokumen n:

$$q_{mk} = \frac{\sum_{n=1}^{N} r_{nk} I(t_m \in d_n)}{\sum_{n=1}^{N} r_{nk}}$$
(2.6)

Probabilitas sebuah klaster k:

$$\alpha_k = \frac{\sum_{n=1}^{N} r_{nk}}{N} \tag{2.7}$$

4. Ulangi langkah 2 dan 3 sampai Convergence. Nilai probabilitas klaster data Convergence jika pengupdetan probabilitas data terhadap klaster data tidak berubah-ubah lagi. Dengan kata lain nilai probabilitas dokumen terhadap sebuah klaster sudah bernilai 1.

Langkah 1: Tentukan nilai threshold. Semakin kecil nilai threshold maka semakin dekat dengan convergence. Dalam hal ini nilai threshold nya adalah nol.

Langkah 2: Hitung nilai Means Square Error dengan menggunakan rumus:

$$MSE(\hat{\theta}) = E[(\hat{\theta} - \theta)^2]_{(2.8)}$$

Langkah 3: Bandingkan Nilai MSE dengan threshold

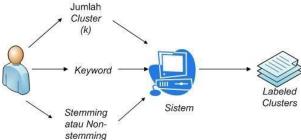
 $Jika^{MSE} \leq Threshold$ maka konvergen dan iterasi berhenti.

3 Metode Penelitian Dan Perancangan 3.1 **Gambaran Umum Sistem**

Sistem yang dibangun adalah sistem yang mengelompokkan dokumen Tugas Akhir Universitas Telkom hasil pencarian pengguna. Untuk proses clustering, sistem menggunakan algoritma Expectation-maximization dengan Naïve Bayes untuk mengelompokkan data tanpa adanya model dari data training dimana pengelompokan ini berdasarkan model parameter yg di random pada tahap sebelum e-step.

Sistem ini akan memproses dokumen yang terdiri dari judul, abstrak dan keyword. Dokumen tersebut akan di kelompokkan berdasarkan abstrak. Dokumen yang di inputkan ke database akan melalui proses preprocessing, dimana outputnya adalah kumpulan term atau kata-kata. Parameterparameter inputan pada proses clustering adalah query, stemming atau tanpa stemming dan jumlah klaster (K). Ketika user melakukan pencarian, maka sistem akan menampilkan dokumen hitlist yang sudah dikelompokkan kedalam beberapa klaster sesuai dengan inputan jumlah klaster dan klaster-klaster tersebut akan diberi label, dimana label tersebut adalah label dokumen yang paling banyak muncul pada suatu klaster.

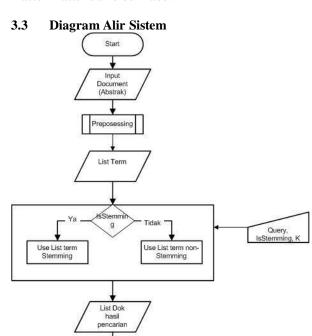
3.2 Arsitektur Sistem



Gambar 3-1Arsitektur Sistem

Pertama user memasukan query search, stemming atau nonstemming, dan jumlah klaster. Sistem kemudian akan memproses parameter input

dan menampilkan dokumen-dokumen hasil pencarian yang sudah dikelompokkan kedalam klaster-klaster dan diberi label.



Gambar 3-2 Diagram Alir Sistem

EM

Pelabelan

cluster

END

a. Preprocessing

Tahap Preprocessing adalah tahap pengubahan dokumen ke dalam bentuk term-term. Tahap ini mempunyai beberapa proses yaitu cleansing, tokenizing, parsing, stopword removal dan stemming.

b. Clustering

Tahap *clustering* adalah tahap pengelompokan dokumen berdasarkan abstraksi. Tahap ini memiliki beberapa proses utama yaitu *Guess Model Parameter, E-Step dan M-Step.*

c. Pelabelan

Setelah proses pengelompokan selesai, maka didapatkan *cluster* yang telah memiliki satu atau lebih anggota. Setiap klaster ini selanjutnya akan diberi nama yang biasa disebut label. Nama label tersebut bertujuan untuk menggambarkan isi dari semua dokumen yang ada dalam tiap klasternya. Nama label ini ditentukan dari jumlah label aktual yang paling banyak pada suatu klaster.

4 Implementasi

Sistem pengelompokan dokumen proses mempunyai empat utama yaitu preprocessing, searching, clustring EM, dan pelabelan. Data yang diinputkan berbentuk teks yang merupakan abstraksi dari data Tugas Akhir S-1 Universitas Telkom. Abstraksi tersebut akan melalui tahap preprocessing yaitu cleansing, tokenizing, parsing, dan stopword elimination (Removal). Kemudian akan dilakukan proses stemming jika user memilih proses stemming (IsStemming="True") dan proses ini akan dilewati memilih proses user tidak (IsStemming="False"). Kemudian melakukan searching dengan menginputkan query, IsStemming dan jumlah klaster.

Dokumen-dokumen yang masuk ke dalam hit list akan di kelompokkan menggunakan algoritma EM kedalam beberapa klaster sesuai dengan input pengguna. Kemudian klaster-klaster tersebut akan dilabeli dengan menggunakan label yg paling banyak muncul pada setiap klaster. Tabel berikut adalah tabel fungsi-fungsi utama yang digunakan pada sistem.

Table 4-1 Tabel Fungsi pada sistem

No.	Proses	Fungsi	Keterangan
1.	Preprocessing	function cleansing()	Fungsi untuk menghilangkan karakter- karakter selain huruf seperti tanda baca, dan simbol
		function tokenizing()	Fungsi untuk mengubah semua huruf dalam dokumen menjadi huruf kecil. Huruf yang diterima hanya huruf "a" sampai dengan "z"
		function parsing()	Fungsi untuk mengubah dokumen menjadi kumpulan kata atau daftar kata (term)
		function stopword()	Fungsi untuk membuang kata-kata yang sering muncul dan tidak memiliki arti deskriptif terhadap isi dokumen. Kata-kata yang termasuk dalam stopwords, misalnya kata "yang", "di",

			"dari" dan		
			sebagainya		
		function	Fungsi untuk		
		stemming()	mencari <i>root</i>		
		stemming()	atau kata dasar		
			dari setiap kata		
			hasil stopwords		
			removal		
2.	Searching	function	Fungsi untuk		
۷.	Searching	search()	mengidentifikasi		
		scarcii()	dokumen-		
			dokumen yang diinginkan oleh		
			user sesuai		
			dengan Query		
			~ .		
	CI.	c	yang diinputkan		
3.	Clustering	function	Fungsi untuk		
	EM	EM()	melakukan		
			pengelompokan		
			dokumen hasil		
			pencarian		
			kedalam		
			beberapa klaster		
4.	Pelabelan	function	Fungsi untuk		
		labelfromDB	memberikan		
		()	label terhadap		
			setiap klaster		
			yang terbentuk		

5 Pengujian Dan Analisis

Pengujian dilakukan dengan cara menghitung nilai performansi algoritma Expectation-maximization dengan menggunakan parameter *recall*, *precision* dan *F-measure* berdasarkan dokumen hasil pengelompokan oleh sistem terhadap pengelompokan secara manual atau kelompok aktual.

Pengujian dibagi menjadi dua, yaitu pengujian parameter untuk data yang telah melalui proses stemming dan data yg tidak melalui proses stemming. Hasil pengelompokan dokumen menggunakan algoritma Expectation-maximization akan berubah-ubah karena penggunaan nilai random pada tahap Guess Model Parameter untuk variabel q_{mk} (Term Probability) dan α_k (Cluster Probability). Oleh karena itu akan dilakukan percobaan sebanyak 5 kali untuk masing-masing query percobaan.

Table 5-1 Tabel Data Pengujian

	Table 5-1 Tabel Data Pengujian								
N	Query	Jumlah	Jumlah	Jumlah	Jumlah				
О		Dokumen	Label	Percobaan	Klaster				
		Hint List	Aktual		(K)				
1	Clusterin	5	2	5	2				
	g								
2	Database	8	3	5	3				
3	Twitter	18	4	5	4				

5.1 Query "Clustering"

Table 5-2 Data Hit List Untuk Query "Clustering"

No	Iddok	Actual Label				
1	7815	Rekayasa Perangkat Lunak				
2	8450	Rekayasa Perangkat Lunak				
3	8974	Rekayasa Perangkat Lunak				
2	3958	Sistem Komputer dan				
		Jaringan Komputer				
3	5634	Sistem Komputer dan Jaringan Komputer				

Berdasarkan 5 kali percobaan untuk masingmasing stemming dan non-stemming, didapatkan nilai recall, precision, dan f-measure sebagai berikut:

Table 5-3 Rata-rata F-measure Query "Clustering"

I ubic 5 .	Clubici					
Percobaan	Precision		Recall		F-Measure	
	Stem	Non- Stem	Stem	Non- Stem	Stem	Non- Stem
1	0.5	1	0.5	1	0.5	1
2	0.5	0.5	0.5	0.75	0.5	0.6
3	1	0.333	1	0.5	1	0.4
4	1	0.5	1	0.5	1	0.5
5	0.5	0.5	0.5	0.5	0.5	0.5
Rata-Rata F-Measure					0.7	0.6

5.2 Query "Database"

Table 5-4 Data Hit List Untuk Query "Databse"

No	Iddok	Actual Label				
1	2868	Image Processing				
2	2920	Image Processing				
3	2967	Image Processing				
4	3482	Image Processing				
5	7836	Informatika Teori dan				
		Pemrograman				
6	8292	Informatika Teori dan				
		Pemrograman				
7	8671	Pengolahan Sinyal				
		Informasi				
8	9051	Informatika Teori dan				
		Pemrograman				

Berdasarkan 5 kali percobaan untuk masingmasing stemming dan non-stemming, didapatkan nilai recall, precision, dan f-measure sebagai berikut:

Table 5-5 Rata-rata F-measure Query "Database"

Table 5-5 Kata-rata F-illeasure Query					Databa	56
Percobaan	Precision		Recall		F-Measure	
	Stem	Non- Stem	Stem	Non- Stem	Stem	Non- Stem
1	0.375	0.636	0.333	0.778	0.353	0.7
2	0.444	0.571	0.444	0.444	0.444	0.5
3	0.857	0.286	0.667	0.222	0.75	0.25
4	0.571	0.444	0.444	0.444	0.5	0.444
5	0.364	0.375	0.444	0.333	0.4	0.353
Rata-Rata F-Measure					0.4894	0.4494

5.3 Query "Clustering"

Table 5-6 Data Hit List Untuk Query "Twitter"

	Table 5-6 Data Hit List Untuk Query "Twitter"						
No	Iddok	Actual Label					
1	238	Branding					
2	268	Branding					
3	2689	Communication					
4	2720	Marketing					
5	2742	Marketing					
6	3081	Marketing					
7	3111	Consumer Behavior And					
		Attitudes					
8	3160	Communication					
9	3166	Communication					
10	3187	Consumer Behavior And					
		Attitudes					
11	3225	Consumer Behavior And					
		Attitudes					
12	3242	Communication					
13	3295	Marketing					
14	3341	Consumer Behavior And					
		Attitudes					
15	3356	Branding					
16	3387	Consumer Behavior And					
		Attitudes					
17	4004	Communication					
18	4092	Branding					

Berdasarkan 5 kali percobaan untuk masingmasing stemming dan non-stemming, didapatkan nilai recall, precision, dan f-measure sebagai berikut:

Table 5-7 Rata-rata F-measure Query "Twitter"

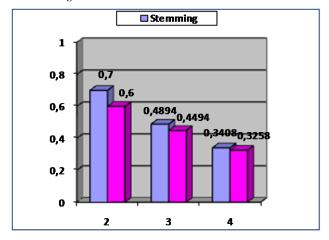
Percobaan	Precision		Recall		F-Measure	
	Stem	Non- Stem	Stem	Non- Stem	Stem	Non- Stem
1	0.246	0.198	0.438	0.5	0.315	0.284
2	0.258	0.304	0.5	0.531	0.34	0.387
3	0.425	0.2	0.531	0.5	0.472	0.286
4	0.265	0.294	0.281	0.313	0.273	0.303
5	0.233	0.364	0.438	0.375	0.304	0.369
Rata-Rata F-Measure					0.3408	0.3258

5.4 Analisis Hasil Penguijan

Pengelompokan dokumen hasil pencarian menggunakan algoritma Expectation-maximization pada sistem ini berguna untuk mengelompokkan dokumen-dokumen yang belum berlabel atau unlabeled documents.

Hasil percobaan pengelompokan dokumen dengan jumlah 5 dokumen dengan jumlah cluster sebanyak 2 cluster memiliki nilai F-measure sebesar 0.7 atau akurasinya sebesar 70% untuk data yang di-stemming dan nilai F-measure sebesar 0.6 atau akurasinya 60% untuk dokumen yang tidak distemming. Namun untuk jumlah dokumen sebanyak 8 dokumen dengan 3 cluster dan 18 dokumen dengan 4 cluster, nilai F-measure menurun menjadi 0.4894 & 0.3408 untuk data yang di-stemming dan

0.4494 & 0.3258 untuk data yang tidak distemming.



Gambar 5-1 Grafik F-measure

Pada gambar diatas dapat dilihat nilai ratarata F-measure yang lebih tinggi untuk data yang melalui proses stemming. Hal ini disebabkan oleh proses stemming yang berfungsi untuk mencari kata dasar sebuah kata berimbuhan. Salah satu sifat algoritma EM yang bekerja secara unsupervised yaitu frekuensi kemunculan kata tidak mempengaruhi proses pengelompokan dokumen. Dengan diubahnya sebuah kata berimbuhan menjadi kata dasar dapat mengurangi jumlah kata atau term yang akan diproses. sehingga peluang kata-kata yang mendeskripsikan isi dari sebuah dokumen dengan kata-kata yang tidak mendeskripsikan isi dari dokumen menjadi seimbang. Misalnya "Clustering" yang muncul sebanyak 10 kali dalam dokumen dengan kata "Percobaan" dan kata "Coba" yang masing-masing muncul 10 kali dalam dokumen, setelah melalui proses stemming peluang kata dasar "Coba" dan "Clustering" akan sama atau seimbang.

Pada pengujian diatas, dilakukan 5 kali percobaan untuk setiap query. Hal ini dikarenakan nilai F-measure yang berubah-ubah karena penggunaan nilai random pada Guess Model Parameter untuk nilai probabilitas awal term (q_{mk}) dan probabilitas awal klaster (α_k) . Tingkat akurasi dari pengelompokan sangat bergantung pada estimasi nilai awal dari q_{mk} dan α_k .

Berdasarkan latar belakang penelitian, dimana dokumen hasil pencarian ditampilkan dengan urutan dokumen atau document ranking, dengan adanya aplikasi pengelompokan dokumen hasil pencarian menggunakan algoritma Expectation-Maximization, dokumen hasil pencarian dikelompokkan sesuai dengan kedekatan dokumen. Dan hal ini dapat membantu pengguna untuk memilih-milah dokumen yang ingin dicari.

6 Kesimpulan dan Saran

6.1 Kesimpulan

Kesimpulan dari hasil implementasi dan analisis algoritma *Clustering Expectation-maximization* pada data Tugas Akhir Universitas Telkom adalah sebagai berikut:

- 1. Berdasarkan latar belakang penelitian, dimana dokumen hasil pencarian ditampilkan dengan document ranking, dengan adanya aplikasi pengelompokan dokumen hasil pencarian menggunakan algoritma Expectation-maximization, maka dokumen hasil pencarian dapat dikelompokkan sesuai dengan kedekatan dokumen. Dan hal ini dapat membantu pengguna untuk memilih-milah dokumen yang ingin dicari.
- 2. Nilai rata-rata akurasi algortima Clustering Expectation-mazimization yang paling tinggi adalah 70% untuk data yang sudah di-stemming dan 60% untuk data yang belum di-stemming. Sedangkan rata-rata akurasi yang paling rendah adalah 34.08% untuk data yang sudah di-stemming dan 32.58% untuk data yang

belum di-*stemming*. Dimana rata-rata akurasi diambil dari 5 kali percobaan untuk setiap *Keyword*.

3. Penambahan proses *stemming* Arifin dan Setiono dapat meningkatkan performansi algoritma *Cluster*ing EM.

6.2 Saran

Untuk meningkatkan performansi algoritma *Cluster*ing *Expectation-maximization* sebaiknya ditambahkan proses *feature selection* dan sinonim kata untuk mengurangi *term-term* yang tidak mencerminkan isi dari sebuah dokumen agar didapatkan hasil yang optimal.

Daftar Pustaka

- [1] Aggarwal, Charu C., ZhaiChengXiang, A Survey of Text Clustering
 Algorithms, Yorktown Heights, NY, IBM T.
 J. Watson Research Center.
- [2] Borman, Sean, 2009, The Expectation Maximization Algorithm A short Tutorial.
- [3] Manning, Christopher D., 2009, An Introduction to Information Retrieval, Cambridge University Press Cambridge, England.
- [4] Liang, Percy, Klein Dan, *Online EM for Unsupervised Models*, University of California at Berkeley, Berkeley, CA 94720.
- [5] Nivre, Joakim, Statistical Methods for NLP Hidden variables and EM, Uppsala University
- [6] Xia, rui, 2013, Naive Bayes for Unsupervised/Semi-supervised Learning via Expectation Maximization(EM).
- [7] Mitchell, Tom m., 2010, Machine Learning. McGraw-Hill Science.
- [8] Andrew Ng, *The EM Algorithm*, Lecture Notes.
- [9] Anneel, Pieter, Maximum Likelihood (ML),

Expectation Maximization (EM), UC Berkeley EECS

[10] Dellaert, Frank, 2002, *The expectation Maximization Algorithm*, Georgia Institute of Technology